2006 IEEE Congress on Evolutionary Computation
Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada
July 16-21, 2006

# Constrained Single-Objective Optimization Using Particle Swarm Optimization

Karin Zielinski and Rainer Laur

***Abstract*— Particle Swarm Optimization (PSO) is an optimization method that is derived from the behavior of social groups like bird flocks or fish schools. In this work PSO is used for the optimization of the constrained test suite of the special session on constrained real parameter optimization at CEC06. Constraint-handling is done by modifying the procedure for determining personal and neighborhood best particles. No additional parameters are needed for the handling of constraints. Numerical results are presented, and statements are given about which types of functions have been successfully optimized and which features present difficulties.**

## I. Introduction

Particle Swarm Optimization (PSO) is a rather new optimization algorithm that was developed in 1995 [1]. PSO was first used for unconstrained single-objective optimization, but in the meantime several approaches for constrained single-objective optimization using PSO have been reported in literature ([2], [3], [4]). In this work constraints are regarded by modifying the selection procedure for personal and neighborhood best individuals. Feasible solutions are always favored over infeasible solutions: If a personal or neighborhood best solution becomes feasible, it cannot get infeasible afterwards. However, the particles themselves are able to switch from feasibility to infeasibility because using PSO every change in the individual is always accepted.

In this work PSO is employed for the optimization of the 24 functions of the test suite for the special session on constrained real parameter optimization at CEC06. Results are presented according to the guidelines in [5]. Furthermore, statements are given about the types of functions which are successfully optimized and which are not.

This paper is organized as follows: At first the Particle Swarm Optimization algorithm and the employed constraint-handling technique are introduced (Section II). The results for the test suite functions are presented in Section III, and conclusions are given in Section IV.

## II. Particle Swarm Optimization

Particle Swarm Optimization imitates the behavior of social groups to solve optimization problems. Although PSO is derived from artificial life, it is usually classified as an evolutionary algorithm because its operators resemble the ones used in evolutionary computation. However, the nomenclature is different from other evolutionary algorithms as the population members are called particles. The particles are characterized by their current position $\vec{x}_i$, their current velocity $\vec{v}_i$ and their personal best position $\vec{p}_i$ (the position that yields the smallest objective function value found so far by the respective particle if minimization problems like the given functions are considered). Positions and velocities are represented as real-valued vectors with dimension $D$ that equals the number of objective function variables.

Karin Zielinski and Rainer Laur are with the Institute for Electromagnetic Theory and Microelectronics (ITEM), University of Bremen, Germany (email: {zielinski,rlaur}@item.uni-bremen.de).

Additionally, the particles possess a memory for the best position $\vec{p}_g$ that has been found so far in a pre-defined neighborhood. Several PSO variants have been developed which vary in the definition of the neighborhood, e.g. the *gbest* or *lbest* neighborhood is in use [6]. In this work the *lbest* variant is employed because it is less susceptible to premature convergence [7]. Using the *lbest* neighborhood, each particle generates a neighborhood containing the particle itself and its two immediate neighbors. The neighbors are defined as the particles with adjacent indices. They are not necessarily in the physical vicinity of the original particle, neither regarding objective function values nor positions.

The algorithm is described by two equations which update the velocity and position of the particles:

$$\vec{v}_{i,G+1} = w\vec{v}_{i,G} + c_1 r_1[\vec{p}_{i,G} - \vec{x}_{i,G}] + c_2 r_2[\vec{p}_{g,G} - \vec{x}_{i,G}] \quad (1)$$

$$\vec{x}_{i,G+1} = \vec{x}_{i,G} + \vec{v}_{i,G+1} \quad (2)$$

The update equation for the velocity (1) is composed of three terms. The first term is linearly dependent on the old velocity. The second term is the so-called cognitive component as it represents the own experience of the particle. The third term is the social component that defines the interaction with the group. The amount of cognitive and social influence is adjusted by the user-defined parameters $c_1$ and $c_2$. The random numbers $r_1$ and $r_2$ are redetermined in the interval [0,1] for each particle in every generation. Another parameter that has to be set by the user is the inertia weight $w$. Other variants of PSO use a constriction coefficient instead, but it is shown in [2] that no fundamentally different results are obtained. Additional parameters of the PSO algorithm are the number of individuals $NP$ and the maximum velocity $V_{max}$ that is used for preventing oscillations with increasing magnitude [8].

### A. Handling of Constraint Functions

Most constrained optimization problems are solved by the application of penalty functions although a lot of different approaches for constraint-handling exist [9]. In this work a method is employed that modifies the procedure of determining personal and neighborhood best solutions [10]. In

0-7803-9487-9/06/$20.00/©2006 IEEE

unconstrained single-objective optimization it is checked for each particle in every generation if a lower objective function value than the personal best solution or the neighborhood best solution has been obtained, and the respective solutions are replaced. Here this procedure is modified, so that a personal or neighborhood best solution $\vec{g}$ is substituted by a new solution $\vec{x}$ if:

- Both vectors are feasible, but $\vec{x}$ yields the smaller objective function value.
- $\vec{x}$ is feasible and $\vec{g}$ is not.
- Both vectors are infeasible, but $\vec{x}$ results in the lower sum of constraint violations.

From this follows that no additional parameters are needed for the applied constraint-handling technique. While an individual is infeasible, only the amount of constraint violations is considered for the evaluation of the particle. As lower constraint violations are preferred, the particles are guided to feasible regions. Because the objective function is not evaluated for infeasible individuals, the algorithm is also suitable for problems where the objective function value cannot be calculated for infeasible individuals. This applies especially to real-world problems.

For unconstrained problems the modified PSO algorithm is exactly the same as the original PSO.

### B. Handling of Boundary Constraints

Although boundary constraints can be seen as constraint functions, they are usually considered separately. A reason is that in real-world problems generally some boundary constraints must not be violated, but for some constraint-handling techniques the value of the objective function is computed for infeasible individuals (e.g. penalty functions).

In this work a limit-exceeding position is set to the middle between old position and limit:

$$u_{j,i,G+1} = \begin{cases} \frac{1}{2} \cdot \left(x_{j,i,G} + x_j^U\right) & \text{if } u_{j,i,G+1} > x_j^U \\ \frac{1}{2} \cdot \left(x_{j,i,G} + x_j^L\right) & \text{if } u_{j,i,G+1} < x_j^L \\ u_{j,i,G+1} & \text{otherwise} \end{cases} \tag{3}$$

where $x_j^U$ is the upper limit and $x_j^L$ is the lower limit. Note that for functions g02 and g14 Equation (3) is slightly modified because here the constraints are of the form $x_j^L < x \leq x_j^U$ instead of $x_j^L \leq x \leq x_j^U$ as for the other functions.

## III. Main Results

### A. PC Configuration

System: Windows XP Professional
CPU: Pentium 4 2.4GHz
RAM: 1GB
Language: C++
Algorithm: Particle Swarm Optimization

### B. Parameter Settings

The following parameters have to be adjusted: $c_1$, $c_2$, $w$, $NP$ and $V_{max}$. The maximum velocity $V_{max}$ is usually set in dependency on the scale of the respective optimization problem [11]. In many cases it is set to the distance between lower and upper limit or to half this range for each dimension ([7], [8], [12]). In this work $V_{max}$ is assigned to one half of the search space in every dimension, respectively.

For the choice of $NP$ not many guidelines exist. In [13] and [14] a population size of 30 is recommended. Most authors choose $20 \leq NP \leq 100$ ([2], [3], [11], [15], [16] [17]). As the number of function evaluations that is needed for convergence generally increases with a higher number of individuals ([11], [13]), but some of the test suite functions have a high dimensionality, $NP = 50$ is used as a trade-off.

Complex interactions exist between $c_1$, $c_2$ and $w$ which have been investigated by different authors ([14], [18], [19]) but general guidelines have not been presented up to now. Most authors use values of $0.5 \leq c_1, c_2 \leq 2$ ([2], [15], [20]) and $0 < w < 1$. The inertia weight $w$ is often linearly decreasing with the number of generations ([2], [15], [17], [21]) but in this work a constant $w = 0.8$ is chosen [22]. Furthermore, $c_1 = 0.5$ and $c_2 = 2.0$ are used, so that the neighborhood best solution is more emphasized than the personal best solution.

For parameter tuning six different combinations of $w$, $c_1$, $c_2$ and $NP$ were examined. Three of the settings were used for five test runs with seven test functions, respectively, while for the other three settings five runs of all 24 test functions were done. The computational cost for parameter tuning in terms of objective function evaluations (FES) is therefore $3 \cdot 5 \cdot 7 \cdot 500,000 + 3 \cdot 5 \cdot 24 \cdot 500,000 = 232,500,000$ FES.

### C. Results

The achieved function error values $(f(\vec{x}) - f(\vec{x}^*))$ for the 24 optimization problems are presented in Tables I, II, III and IV ($f(\vec{x}^*)$ is the best known solution from [5]). The best, median, worst and mean solution and the standard deviation are given for 5000, 50000 and 500000 FES, respectively. All numbers are computed from 25 optimization runs. For the best, median and worst results the number of violated constraints is also given (see numbers in brackets). For the median solution this statement is more accurately specified by presenting the number of constraints $c$ that are violated by more than 1, 0.01 and 0.0001 and furthermore the mean constraint violations $\bar{v}$ [5].

The numbers of function evaluations that are necessary for satisfying the condition (4) are shown in Table V.

$$f(\vec{x}) - f(\vec{x}^*) \leq 0.0001 \text{ with feasible } \vec{x} \tag{4}$$

Again best, median, worst and mean results as well as the standard deviation are presented.

Other information given in Table V is the feasible rate, the success rate and the success performance. The feasible rate represents the percentage of runs for which at least one feasible solution was found, and the success rate is the percentage of runs for which condition (4) holds. The success performance is calculated by multiplying the mean number of function evaluations by the number of total runs and dividing the result by the number of successful runs [5].

For all but three functions a feasible rate of 100% has been achieved (see Table V). The exceptions are g21, for which a

TABLE I

ERROR VALUES ACHIEVED WHEN FES= $5 \times 10^3$ , FES= $5 \times 10^4$, FES= $5 \times 10^5$ FOR PROBLEMS 1-6.

| FES | | g01 | g02 | g03 | g04 | g05 | g06 |
|---|---|---|---|---|---|---|---|
| $5 \times 10^3$ | Best | 7.6099E-01(0) | 3.0145E-01(0) | 9.0284E-01(0) | 5.4482E-01(0) | 4.4918E+01(3) | 1.6223(0) |
| | Median | 4.6293(0) | 4.6694E-01(0) | 1.0005(0) | 4.3440(0) | 9.8691E+02(3) | 1.8373E+01(0) |
| | Worst | 7.5932(0) | 5.3468E-01(0) | 1.0005(0) | 6.7157E+01(0) | -5.3836(4) | 3.0130E+02(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 2,3,3 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 9.8781E-01 | 0 |
| | Mean | 4.6075 | 4.5183E-01 | 9.9325E-01 | 8.4928 | 4.5938E+02 | 4.4837E+01 |
| | Std | 1.8292 | 6.2101E-02 | 2.0970E-02 | 1.3495E+01 | 4.2775E+02 | 6.4625E+01 |
| $5 \times 10^4$ | Best | 1.1575E-09(0) | 1.0498E-01(0) | 8.9457E-01(0) | 7.6398E-11(0) | 1.3148E-03(0) | 4.5475E-11(0) |
| | Median | 3.0000(0) | 2.6692E-01(0) | 1.0003(0) | 7.6398E-11(0) | 4.3680E+02(0) | 4.5475E-11(0) |
| | Worst | 6.0000(0) | 4.5896E-01(0) | 1.0005(0) | 7.6761E-10(0) | 5.1232E+02(1) | 4.5475E-11(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 2.0449 | 2.6339E-01 | 9.8953E-01 | 1.1729E-10 | 4.4443E+02 | 4.5475E-11 |
| | Std | 1.8993 | 7.6134E-02 | 2.7266E-02 | 1.3805E-10 | 3.8546E+02 | 5.0077E-19 |
| $5 \times 10^5$ | Best | 0(0) | 1.0330E-01(0) | 5.9072E-01(0) | 7.6398E-11(0) | -9.0949E-13(0) | 4.5475E-11(0) |
| | Median | 0(0) | 2.6450E-01(0) | 9.6347E-01(0) | 7.6398E-11(0) | 1.3016E+01(0) | 4.5475E-11(0) |
| | Worst | 6.0000(0) | 4.2046E-01(0) | 9.9905E-01(0) | 7.6398E-11(0) | 6.1794E+02(0) | 4.5475E-11(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 1.4938 | 2.4286E-01 | 9.0719E-01 | 7.6398E-11 | 1.0585E+02 | 4.5475E-11 |
| | Std | 1.8926 | 8.2491E-02 | 1.2203E-01 | 0 | 1.7089E+02 | 5.0077E-19 |

TABLE II

ERROR VALUES ACHIEVED WHEN FES= $5 \times 10^3$ , FES= $5 \times 10^4$, FES= $5 \times 10^5$ FOR PROBLEMS 7-12.

| FES | | g07 | g08 | g09 | g10 | g11 | g12 |
|---|---|---|---|---|---|---|---|
| $5 \times 10^3$ | Best | 1.6781E+02(0) | 6.4245E-10(0) | 1.6366E+01(0) | 3.6177E+03(0) | 2.4501E-04(0) | 2.3327E-07(0) |
| | Median | 3.4160E+02(0) | 6.0932E-08(0) | 5.3381E+01(0) | 9.8517E+03(0) | 4.1574E-02(0) | 3.8738E-05(0) |
| | Worst | 1.4245E+03(1) | 2.8172E-06(0) | 1.3230E+02(0) | 1.7798E+04(1) | 1.6707E-01(0) | 8.1848E-03(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 5.7280E+02 | 3.1300E-07 | 5.7713E+01 | 9.0457E+03 | 6.6153E-02 | 1.6678E-03 |
| | Std | 4.9105E+02 | 6.6313E-07 | 2.9574E+01 | 3.9916E+03 | 5.9456E-02 | 2.9365E-03 |
| $5 \times 10^4$ | Best | 4.1413E-01(0) | 8.1964E-11(0) | 9.0772E-06(0) | 1.9584E+01(0) | 0(0) | 0(0) |
| | Median | 1.3074(0) | 8.1964E-11(0) | 4.0110E-04(0) | 3.2682E+02(0) | 0(0) | 0(0) |
| | Worst | 2.6532(0) | 8.1964E-11(0) | 8.0374E-03(0) | 1.2834E+03(0) | 4.7962E-14(0) | 0(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 1.4466 | 8.1964E-11 | 1.5148E-03 | 4.2244E+02 | 2.1894E-15 | 0 |
| | Std | 6.3268E-01 | 6.6435E-18 | 2.2555E-03 | 3.0401E+02 | 9.5553E-15 | 0 |
| $5 \times 10^5$ | Best | 2.7592E-05(0) | 8.1964E-11(0) | -9.8112E-11(0) | 9.8562E-09(0) | 0(0) | 0(0) |
| | Median | 1.0287E-02(0) | 8.1964E-11(0) | -9.7884E-11(0) | 6.0547E-04(0) | 0(0) | 0(0) |
| | Worst | 7.6104E-01(0) | 8.1964E-11(0) | -9.6634E-11(0) | 1.2829E+03(0) | 0(0) | 0(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 1.9395E-01 | 8.1964E-11 | -9.7702E-11 | 5.3303E+01 | 0 | 0 |
| | Std | 2.8742E-01 | 1.0015E-18 | 4.3166E-13 | 2.5635E+02 | 0 | 0 |

feasible rate of $8\%$ is calculated, as well as g20 and g22 for which no feasible individual at all could be identified.

Success rates of $100\%$ have been accomplished for eight functions (see Table V). For two other functions the success rate is rather high ($80\%$), for five functions the success rate is between $52\%$ and $8\%$ while for nine functions the best known solutions from [5] have not been found.

In the following general statements are derived about the types of functions which are successfully optimized or which create problems here. Information that exceeds the statements of [5] is given in [23] for g01-g13 and in [24] for g14-g24.

The functions for which success rates of $100\%$ have been reached all have a rather low dimensionality of $2 \leq D \leq 7$. The dimensionality of the two functions with success rates of $80\%$ is also moderate ($D \leq 9$). Therefore, it is suspected that a high dimensionality presents difficulties. To verify if a larger population size is able to avoid these problems, the parameter setting $NP = 100$ is examined. It can be seen in Table VI that for six test problems (g01, g05, g07, g15, g18, g19) the success rate improves (for all but one of these functions the success performance also improves). For six problems with feasible rates of $100\%$ and success rates of $0\%$ no different results are obtained (g02, g03, g13, g14, g17, g23). For the two problems with feasible rates of $0\%$ the results are also identical (g20, g22). There is one function for which worse results are obtained (g10). To sum up, a

TABLE III

ERROR VALUES ACHIEVED WHEN FES= $5 \times 10^3$ , FES= $5 \times 10^4$, FES= $5 \times 10^5$ FOR PROBLEMS 13-18.

| FES | | g13 | g14 | g15 | g16 | g17 | g18 |
|---|---|---|---|---|---|---|---|
| $\mathbf{5 \times 10^3}$ | Best | 6.3301E-01(2) | 9.2057(2) | 3.2328(0) | 2.6891E-02(0) | 1.3173E+02(4) | 5.7799E-01(0) |
| | Median | 9.3306E-01(3) | 6.3248(3) | 5.7840(2) | 6.3493E-02(0) | 1.0342E+02(4) | 1.4714(9) |
| | Worst | 1.0604(3) | 1.2471E+01(3) | 5.6675(2) | 1.5135E-01(0) | 1.2988E+02(4) | -9.1216E-01(7) |
| | $c$ | 0,3,3 | 0,0,3 | 0,1,2 | 0,0,0 | 3,4,4 | 0,9,9 |
| | $\overline{v}$ | 6.6850E-02 | 5.8034E-03 | 6.3607E-03 | 0 | 3.4917 | 3.0659E-01 |
| | Mean | 9.5815E-01 | 8.7551 | 4.1640 | 6.5822E-02 | 1.5880E+02 | 8.3177E-01 |
| | Std | 5.3289E-01 | 3.2332 | 3.0131 | 2.7580E-02 | 6.7238E+01 | 5.4913E-01 |
| $\mathbf{5 \times 10^4}$ | Best | 7.8017E-02(0) | 3.9103E-01(0) | 8.0125E-06(0) | 2.8554E-07(0) | 9.7730E+01(0) | 4.4626E-03(0) |
| | Median | 8.5607E-01(0) | 5.5091(0) | 8.9491E-02(0) | 1.9309E-06(0) | 1.5762E+02(3) | 3.2903E-02(0) |
| | Worst | 1.1146(0) | 9.1244(0) | 1.8528(0) | 2.7723E-04(0) | 1.5838E+02(4) | 3.4570E-01(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,3 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 2.3671E-03 | 0 |
| | Mean | 7.4665E-01 | 5.3105 | 3.7717E-01 | 1.5694E-05 | 1.5368E+02 | 6.5290E-02 |
| | Std | 2.4902E-01 | 2.5562 | 5.3383E-01 | 5.5190E-05 | 4.9825E+01 | 8.7153E-02 |
| $\mathbf{5 \times 10^5}$ | Best | 4.5154E-02(0) | 2.0885E-01(0) | 6.0822E-11(0) | 6.5214E-11(0) | 3.3785E+01(0) | 1.5561E-11(0) |
| | Median | 5.4335E-01(0) | 4.6345(0) | 6.1277E-11(0) | 6.5214E-11(0) | 1.1363E+02(0) | 1.6160E-11(0) |
| | Worst | 9.0850E-01(0) | 8.5490(0) | 1.1429(0) | 6.5214E-11(0) | 1.5855E+02(0) | 2.5299E-01(0) |
| | $c$ | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 0 | 0 | 0 | 0 | 0 |
| | Mean | 5.1868E-01 | 4.2588 | 4.8916E-02 | 6.5214E-11 | 1.0788E+02 | 1.2460E-02 |
| | Std | 2.6321E-01 | 2.3316 | 2.2820E-01 | 0 | 4.3374E+01 | 5.0815E-02 |

TABLE IV

ERROR VALUES ACHIEVED WHEN FES= $5 \times 10^3$ , FES= $5 \times 10^4$, FES= $5 \times 10^5$ FOR PROBLEMS 19-24.

| FES | | g19 | g20 | g21 | g22 | g23 | g24 |
|---|---|---|---|---|---|---|---|
| $\mathbf{5 \times 10^3}$ | Best | 2.5822E+01(0) | -2.7276E-02(12) | 8.0628E+02(4) | 1.9178E+04(17) | 4.0005E+02(0) | 1.3464E-05 |
| | Median | 6.4308E+01(0) | 1.0816E-01(12) | -1.9372E+02(5) | -2.3640E+02(20) | 4.0006E+02(0) | 1.1399E-03 |
| | Worst | 6.0048E+02(0) | 4.3236E-01(11) | -1.9372E+02(6) | 1.8480E+04(18) | -1.6999E+03(2) | 4.6151E-03 |
| | $c$ | 0,0,0 | 0,4,7 | 0,3,4 | 19,20,20 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 8.4688E-02 | 1.2190E-01 | 1.1937E+06 | 0 | 0 |
| | Mean | 1.2873E+02 | 4.2983E-01 | 4.6688E+02 | 1.1505E+04 | 2.6806E+02 | 1.5523E-03 |
| | Std | 1.5065E+02 | 1.0987 | 3.7965E+02 | 8.2105E+03 | 4.3657E+02 | 1.2577E-03 |
| $\mathbf{5 \times 10^4}$ | Best | 1.3843(0) | 4.8517E-02(15) | 8.0581E+02(2) | 1.2655E+04(19) | 4.0005E+02(0) | 4.6736E-12 |
| | Median | 8.2040(0) | 1.7768E-01(9) | 3.3578E+02(5) | 1.7476E+04(19) | 4.0006E+02(0) | 4.6736E-12 |
| | Worst | 2.9702E+01(0) | 4.2557E-01(9) | 7.0564E+02(5) | 1.9764E+04(19) | 3.7079E+01(3) | 4.6736E-12 |
| | $c$ | 0,0,0 | 0,2,2 | 0,4,5 | 17,18,19 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 3.5807E-02 | 1.8747E-02 | 3.2012E+05 | 0 | 0 |
| | Mean | 1.1119E+01 | 2.3120E-01 | 6.0815E+02 | 1.0441E+04 | 3.5730E+02 | 4.6736E-12 |
| | Std | 8.4354 | 4.2734E-01 | 2.0972E+02 | 8.2164E+03 | 1.3460E+02 | 8.8525E-20 |
| $\mathbf{5 \times 10^5}$ | Best | 4.7415E-11(0) | -1.3604E-03(6) | 2.9751E+02(0) | 1.8210E+04(19) | 3.0005E+02(0) | 4.6736E-12 |
| | Median | 4.9946E-01(0) | 5.9518E-02(4) | 8.0628E+02(1) | 1.2742E+04(19) | 4.0005E+02(0) | 4.6736E-12 |
| | Worst | 2.6844(0) | 1.0283(9) | 8.0627E+02(1) | 1.9764E+04(9) | 4.0006E+02(0) | 4.6736E-12 |
| | $c$ | 0,0,0 | 0,1,1 | 0,0,1 | 18,19,19 | 0,0,0 | 0,0,0 |
| | $\overline{v}$ | 0 | 2.5031E-02 | 1.5716E-03 | 1.2977E+05 | 0 | 0 |
| | Mean | 6.7306E-01 | 1.8801E-01 | 6.7190E+02 | 1.5677E+04 | 3.7784E+02 | 4.6736E-12 |
| | Std | 7.7373E-01 | 4.3788E-01 | 2.2106E+02 | 7.1412E+03 | 4.0467E+01 | 8.8525E-20 |

higher number of individuals is generally beneficial for the test problems with a high dimensionality but not for all of these functions better results have been found.

The statement of [11] and [13], that a larger population size leads to a higher number of function evaluations for convergence, is confirmed since for all functions with success rate of $100\%$ the success performance is worse with $N = 100$ than with $NP = 50$.

Disjoint feasible regions provide no problems for the applied algorithm since functions g12 and g24 are successfully optimized.

The number of constraints for successfully optimized functions is mostly rather small with the exception of g16 that contains 38 inequality constraints. Besides, function g18, that resulted in a success rate of $80\%$, includes 13 inequality constraints. Therefore, it is concluded that the number of inequality constraints does not create difficulties in general (this result corresponds with the findings in [6]).

The size of the feasible space is not generally a problem, either, because in [5] the ratio between feasible region and search space ($\rho = |F|/|S|$) is given as $\rho = 0.0000\%$ for g11, and g11 has been reliably optimized here.

Furthermore, the successfully optimized functions include quadratic, cubic, nonlinear, polynomial and linear objective functions (this is also in accordance with [24]), thus the type of objective function is not of importance for successful

TABLE V

NUMBER OF FES TO ACHIEVE THE FIXED ACCURACY LEVEL ($(f(\vec{x}) - f(\vec{x}*)) \leq 0.0001$), SUCCESS RATE, FEASIBLE RATE AND SUCCESS PERFORMANCE.

| Prob. | Best | Median | Worst | Mean | Std | Feasible Rate | Success Rate | Success Performance |
|---|---|---|---|---|---|---|---|---|
| g01 | 25273 | 46405 | 346801 | 76195 | 89607 | 100% | 52% | 146530 |
| g02 | - | - | - | - | - | 100% | 0% | - |
| g03 | - | - | - | - | - | 100% | 0% | - |
| g04 | 15363 | 19681 | 25776 | 20546 | 2415 | 100% | 100% | 20546 |
| g05 | 94156 | 440153 | 482411 | 364218 | 181153 | 100% | 16% | 2276363 |
| g06 | 16794 | 20007 | 22274 | 20043 | 1487 | 100% | 100% | 20043 |
| g07 | 315906 | 327283 | 338659 | 327283 | 16089 | 100% | 8% | 4091031 |
| g08 | 1395 | 2311 | 3921 | 2360 | 562 | 100% | 100% | 2360 |
| g09 | 45342 | 57690 | 84152 | 58129 | 8671 | 100% | 100% | 58129 |
| g10 | 290367 | 461422 | 486655 | 426560 | 69367 | 100% | 32% | 1332999 |
| g11 | 5475 | 17487 | 21795 | 16386 | 4306 | 100% | 100% | 16386 |
| g12 | 1409 | 3933 | 9289 | 4893 | 2657 | 100% | 100% | 4893 |
| g13 | - | - | - | - | - | 100% | 0% | - |
| g14 | - | - | - | - | - | 100% | 0% | - |
| g15 | 17857 | 171173 | 348138 | 176827 | 108370 | 100% | 80% | 221033 |
| g16 | 24907 | 33021 | 51924 | 33335 | 5786 | 100% | 100% | 33335 |
| g17 | - | - | - | - | - | 100% | 0% | - |
| g18 | 85571 | 177989 | 455907 | 191220 | 96308 | 100% | 80% | 239026 |
| g19 | 307044 | 365284 | 423523 | 365284 | 82363 | 100% | 8% | 4566044 |
| g20 | - | - | - | - | - | 0% | 0% | - |
| g21 | - | - | - | - | - | 8% | 0% | - |
| g22 | - | - | - | - | - | 0% | 0% | - |
| g23 | - | - | - | - | - | 100% | 0% | - |
| g24 | 3547 | 7487 | 9388 | 7262 | 1257 | 100% | 100% | 7262 |

TABLE VI

NUMBER OF FES TO ACHIEVE THE FIXED ACCURACY LEVEL ($(f(\vec{x}) - f(\vec{x}*)) \leq 0.0001$), SUCCESS RATE, FEASIBLE RATE AND SUCCESS PERFORMANCE WITH $NP = 100$.

| Prob. | Best | Median | Worst | Mean | Std | Feasible Rate | Success Rate | Success Performance |
|---|---|---|---|---|---|---|---|---|
| g01 | 47238 | 66935 | 154901 | 73314 | 29004 | 100% | 72% | 101825 |
| g02 | - | - | - | - | - | 100% | 0% | - |
| g03 | - | - | - | - | - | 100% | 0% | - |
| g04 | 28869 | 39046 | 51671 | 37802 | 6135 | 100% | 100% | 37802 |
| g05 | 202186 | 404214 | 470564 | 366824 | 113752 | 100% | 24% | 1528433 |
| g06 | 33819 | 37317 | 42845 | 37946 | 3075 | 100% | 100% | 37946 |
| g07 | 343793 | 394016 | 489448 | 405156 | 39935 | 100% | 72% | 562717 |
| g08 | 1347 | 3808 | 5796 | 3656 | 920 | 100% | 100% | 3656 |
| g09 | 86267 | 101398 | 131641 | 103677 | 8993 | 100% | 100% | 103677 |
| g10 | 487301 | 487525 | 487748 | 487525 | 316 | 100% | 8% | 6094056 |
| g11 | 11155 | 34436 | 43250 | 33073 | 5661 | 100% | 100% | 33073 |
| g12 | 2568 | 6631 | 13282 | 6906 | 2754 | 100% | 100% | 6906 |
| g13 | - | - | - | - | - | 100% | 0% | - |
| g14 | - | - | - | - | - | 100% | 0% | - |
| g15 | 40558 | 294582 | 470020 | 267821 | 142870 | 100% | 84% | 318834 |
| g16 | 40523 | 57022 | 69332 | 56612 | 8387 | 100% | 100% | 56612 |
| g17 | - | - | - | - | - | 100% | 0% | - |
| g18 | 115761 | 224633 | 435978 | 238706 | 63910 | 100% | 100% | 238706 |
| g19 | 417801 | 426887 | 433614 | 426101 | 7936 | 100% | 12% | 3550839 |
| g20 | - | - | - | - | - | 0% | 0% | - |
| g21 | - | - | - | - | - | 8% | 0% | - |
| g22 | - | - | - | - | - | 0% | 0% | - |
| g23 | - | - | - | - | - | 100% | 0% | - |
| g24 | 9833 | 13621 | 18382 | 13791 | 1986 | 100% | 100% | 13791 |

optimization with the applied algorithm.

For functions with a high number of equality constraints (g20 and g22) particularly bad results have been yielded: The feasible rate equals 0% for both functions. The functions with the next highest number of equality constraints are g21 and g17 for which feasible individuals have been found but no successful run has been accomplished. It is concluded from these results that a high number of equality constraints provides a challenge for the applied algorithm.

The highest number of active constraints at the optimum for a function with success rate of $100\%$ is four (g16), and for g10 with six active constraints the success rate is $32\%$. Most functions with an optimum at the boundary between feasible and infeasible region like g01, g07, g10, g18, g20, g21, g22 and g23 also include the difficulty of a rather high dimensionality. For three of these functions better results are obtained with a higher number of individuals, so it is inferred that at least six active constraints at the optimum are not generally a problem. For higher numbers no determination can be made since this applies only to g20 and g22, but these functions also contain another source of difficulty (a high number of equality constraints).

In Table VII the functions are given for which better than the previous best known solutions from [5] have been found. The previous best solutions, the newly found best solutions and the respective parameter settings are displayed. Note that the solutions for g06 and g15 have been found with a setting of $NP = 100$.

TABLE VII

BETTER SOLUTIONS THAN THE GIVEN BEST KNOWN SOLUTIONS [5].

| Prob. | former f(x*) | f(x*) | Parameter Values |
|---|---|---|---|
| g05 | 5126.4967140071 | 5126.4967140070994 | 6.7994514299135858e+002<br>1.0260669816693023e+003<br>1.1887637287791658e-001<br>-3.9623348341934234e-001 |
| g06 | -6961.81387558015 | -6961.8138755801547 | 1.4094999999999994e+001<br>8.4296078921546436e-001 |
| g09 | 680.630057374402 | 680.63005737440187 | 2.3304993583319522e+000<br>1.9513723736921014e+000<br>-4.7754148383125650e-001<br>4.3657262412626956e+000<br>-6.2448699243235684e-001<br>1.0381310559643508e+000<br>1.5942267273331172e+000 |
| g15 | 961.715022289961 | 961.71502228996087 | 3.5121281620752844e+000<br>2.1698750757115579e-001<br>3.5521785139144879e+000 |

### D. Convergence Map

Figures 1, 2, 3 and 4 present convergence graphs for the given optimization problems based on the median of 25 runs, respectively. The function error values as well as the mean constraint violations are given. For the y-axis a logarithmic scale is used. Note that some of the graphs will become too complex if all data points are marked with a symbol, so only every 5000th or every 500th data point is displayed with a symbol, respectively. Furthermore, the range of the y-axis is restricted to $[10^{-10}, 10^5]$ for comparability reason in Figures 1(a), 2(a), 3(a) and 4(a). For the same reason the x-axis is assigned the range $[0, 500000]$ in all figures.

Note that upward jumps in the error function value occur (e.g. for function g20, see Figure 4(a)) if the respective solution is not feasible, because in this case the individual is evaluated only on basis of the amount of constraint violations, and the objective function value is not taken into account. This is advantageous for real-world problems for which the objective function cannot be evaluated for infeasible individuals. However, here this property leads to a degraded performance in terms of the function error value for the functions for which no feasible individuals have been found.

Gaps in the function error value curve (e.g. g20, see Figure 4(a)) happen if the function error values are negative during this time, so they cannot be displayed due to the logarithmic scale of the y-axis.

### E. Algorithm Complexity

In Table VIII an estimate for the algorithm complexity is given in the form $(T2 - T1)/T1$. The numbers $T1$ and $T2$ are computed as follows:

- $T1$ is the average computing time of $10,000$ function evaluations for each optimization problem, respectively: $T1 = (\sum_{i=1}^{24} t1_i)/24$ where $t1_i$ is the computing time for $10,000$ evaluations of problem $i$.
- $T2$ is the average of the computing time for the algorithm if $10,000$ evaluations per optimization problem are calculated: $T2 = (\sum_{i=1}^{24} t2_i)/24$ where $t2_i$ is the computing time for the algorithm with $10,000$ evaluations of optimization problem $i$.

TABLE VIII

COMPUTATIONAL COMPLEXITY

| $T1$ | $T2$ | $(T2 - T1)/T1$ |
|---|---|---|
| 0.1165s | 0.761083s | 5.5329 |

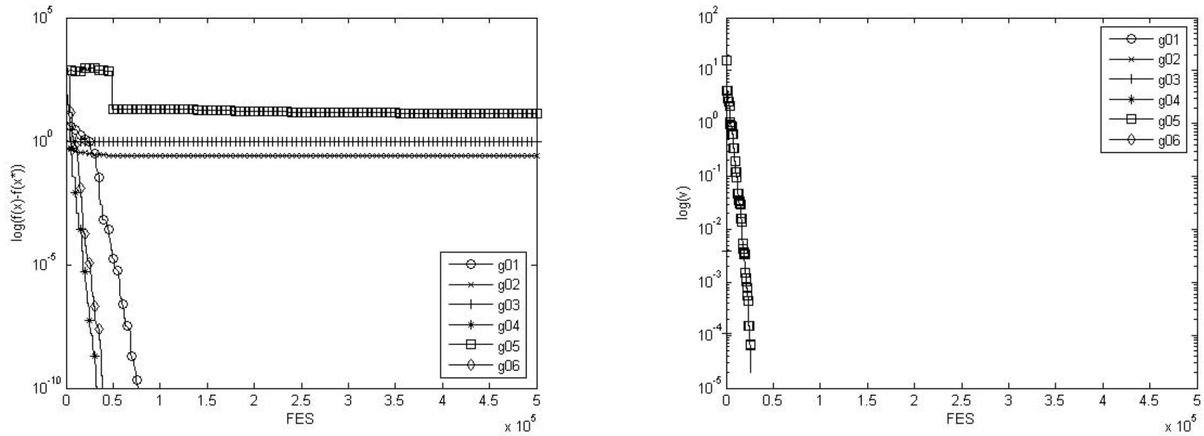
## IV. CONCLUSIONS

In this work the Particle Swarm Optimization algorithm is used for the optimization of 24 constrained single-objective test problems. Instead of using penalty functions for constraint-handling what is the most common approach, a method based on a modified procedure for choosing personal and neighborhood best solutions is used. Therefore, no additional parameters are needed.

Feasible rates of $100\%$ have been yielded for all but three test problems. Success rates of $100\%$ have been accomplished for eight functions. For nine functions the best known solutions have not been found, and the remaining seven functions result in success rates between $8\%$ and $80\%$. For four functions better solutions than the previous best known solutions have been found.

If no feasible individual is found for an optimization problem, the search is guided only by the amount of constraint violations, and the objective function value is not taken into account. This behavior is often favorable for real-world problems where it may be impossible to evaluate the objective function for an infeasible individual, e.g. due to physical limits. However, for some of the provided test functions this property leads to a degraded performance regarding the function error value.

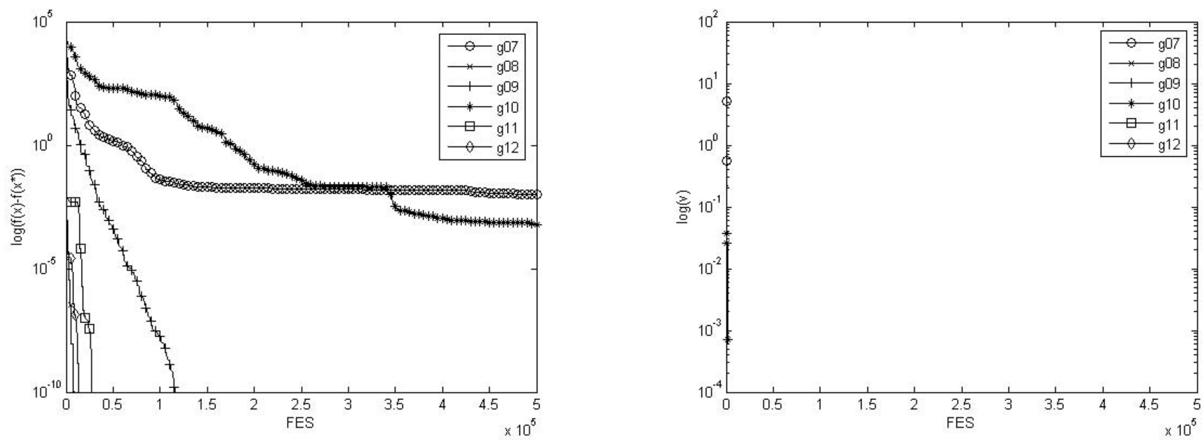
Test functions with disjoint feasible regions, problems with many inequality constraints and different types of objective functions have been successfully optimized. The size of the feasible space in dependence on the size of the search space does not seem to create problems, either.




(a) Function Error Value

(b) Mean Constraint Violations

Fig. 1. Convergence Graph for Problems 1-6




(a) Function Error Value

(b) Mean Constraint Violations

Fig. 2. Convergence Graph for Problems 7-12

The most difficulties are caused by test problems with a high number of equality constraints. Another challenge is a high dimensionality although it is shown that a higher number of individuals yields better results. The effect of a high number of active constraints at the optimum could not be sufficiently analyzed because the test functions with the highest number of active constraints (16 and 19, respectively) also contain a large number of equality constraints. Problems with six active constraints mostly have a high dimensionality for the given test suite, so generally the number of individuals had to be increased for successful optimization.

## References


[1] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, 1995.

[2] K. Parsopoulos and M. Vrahatis, "Particle Swarm Optimization Method for Constrained Optimization Problems," in *Intelligent Technologies - Theory and Applications: New Trends in Intelligent Technologies*, P. Sincak, J. Vascak, V. Kvasnicka, and J. Pospichal, Eds. IOS Press, 2002, pp. 214–220.

[3] X. Hu and R. C. Eberhart, "Solving Constrained Nonlinear Optimization Problems with Particle Swarm Optimization," in *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Informatics*, 2002.

[4] B. Brandstätter and U. Baumgartner, "Particle Swarm Optimization - Mass-Spring System Analogon," *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 997–1000, 2002.

[5] J. Liang, T. P. Runarsson, E. Mezura-Montes, M. Clerc, P. Suganthan, C. A. Coello Coello, and K. Deb, "Problem Definitions and Evaluation Criteria for the CEC 2006 Special Session on Constrained Real-Parameter Optimization," Nanyang Technological University, Tech. Rep., March 2006.

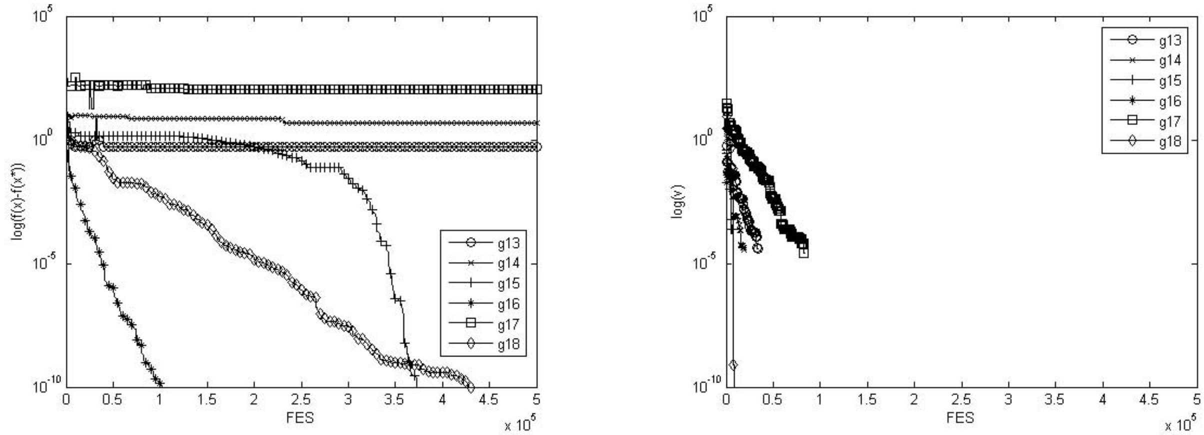
[6] R. Mendes, J. Kennedy, and J. Neves, "The Fully Informed Particle Swarm: Simpler, Maybe Better," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 204–210, 2004.

[7] F. van den Bergh, "An Analysis of Particle Swarm Optimizers," Ph.D. dissertation, University of Pretoria, 2001.

[8] J. Kennedy and R. C. Eberhart, *Swarm Intelligence*. San Francisco: Morgan Kaufmann Publishers, 2001.

[9] C. A. Coello Coello, "Theoretical and Numerical Constraint-Handling Techniques used with Evolutionary Algorithms: A Survey of the State of the Art," *Computer Methods in Applied Mechanics and Engineering*, vol. 191, no. 11-12, pp. 1245–1287, 2002.

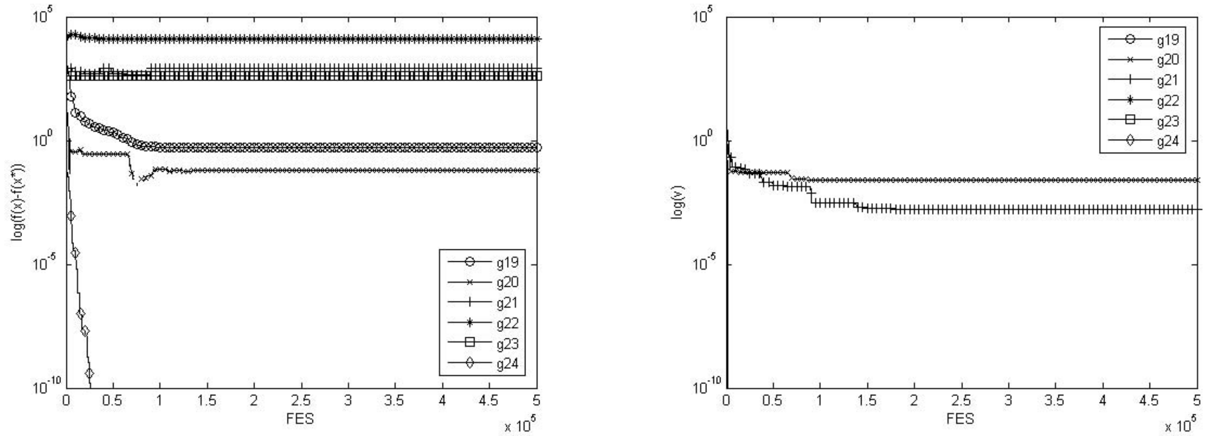
[10] G. T. Pulido and C. A. Coello Coello, "A Constraint-Handling Mechanism for Particle Swarm Optimization," in *Proceedings of the Congress*




(a) Function Error Value

(b) Mean Constraint Violations

Fig. 3. Convergence Graph for Problems 13-18




(a) Function Error Value

(b) Mean Constraint Violations

Fig. 4. Convergence Graph for Problems 19-24


*on Evolutionary Computation*, 2004.

[11] J. Kennedy and R. C. Eberhart, "The Particle Swarm: Social Adaptation in Information-Processing Systems," in *New Ideas in Optimization*, D. Corne, M. Dorigo, and F. Glover, Eds. London: McGraw-Hill, 1999, pp. 379–387.

[12] M. Løvbjerg, T. K. Rasmussen, and T. Krink, "Hybrid Particle Swarm Optimiser with Breeding and Subpopulations," in *Proceedings of the Genetic and Evolutionary Computation Conference*, 2001.

[13] A. Carlisle and G. Dozier, "An Off-The-Shelf PSO," in *Proceedings of the Workshop on Particle Swarm Optimization*, 2001.

[14] I. C. Trelea, "The Particle Swarm Optimization Algorithm: Convergence Analysis and Parameter Selection," *Information Processing Letters*, vol. 85, no. 6, pp. 317–325, 2003.

[15] K. Parsopoulos, V. Plagianakos, G. Magoulas, and M. Vrahatis, "Stretching Technique for Obtaining Global Minimizers Through Particle Swarm Optimization," in *Proceedings of the Workshop on Particle Swarm Optimization*, 2000.

[16] T. Bartz-Beielstein, K. E. Parsopoulos, and M. N. Vrahatis, "Design and Analysis of Optimization Algorithms using Computational Statistics," *Applied Numerical Analysis & Computational Mathematics (ANACM)*, vol. 1, no. 3, pp. 413–433, 2004.

[17] R. Eberhart and Y. Shi, "Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization," in *Proceedings of the Congress on Evolutionary Computation*, 2000.

[18] T. Beielstein, K. E. Parsopoulos, and M. N. Vrahatis, "Tuning PSO Parameters Through Sensitivity Analysis," University of Dortmund, Tech. Rep., 2002.

[19] M. Clerc and J. Kennedy, "The Particle Swarm - Explosion, Stability, and Convergence in a Multidimensional Complex Space," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2002.

[20] X.-F. Xie, W.-J. Zhang, and D.-C. Bi, "Optimizing Semiconductor Devices by Self-Organizing Particle Swarm," in *IEEE Congress on Evolutionary Computation*, 2004.

[21] T. Krink, J. S. Vesterstrøm, and J. Riget, "Particle Swarm Optimisation with Spatial Particle Extension," in *Proceedings of the IEEE Congress on Evolutionary Computation*, 2002.

[22] K. Zielinski, D. Peters, and R. Laur, "Stopping Criteria for Single-Objective Optimization," in *Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems*, 2005.

[23] E. Mezura-Montes and C. A. Coello Coello, "A Simple Multimembered Evolution Strategy to Solve Constrained Optimization Problems," *IEEE Transactions on Evolutionary Computation*, vol. 9, no. 1, pp. 1–17, 2005.

[24] ——, "What Makes a Constrained Problem Difficult to Solve by an Evolutionary Algorithm," Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, Mexico, Tech. Rep., 2004.